194-00131

EOSDIS Core System Project

Defining the Architectural Development Of EOSDIS to Facilitate Extension to a Wider Data Information System

Technical Paper

April 1994

Hughes Applied Information Systems
Landover, Maryland

DEFINING THE ARCHITECTURAL DEVELOPMENT OF EOSDIS TO FACILITATE EXTENSION TO A WIDER DATA INFORMATION SYSTEM

Mark Elkington[§], Richard Meyer[¶] and Gail McConaughy[¥]

[§] *Earth Observation Science Ltd, Hughes Team ECS Project Office, 1616 McCormick Drive, Landover MD 20785*[*]*;*
[¶] *D&M Associates, Hughes Team ECS Project Office, 1616 McCormick Drive, Landover MD 20785*[*]*;*
[¥] *NASA Goddard Space Flight Center, ESDIS Project Office - Code 500, Greenbelt, Maryland*

Commission II, Working Group 2

KEYWORDS: Global Change, Information Systems, Earth Observation

ABSTRACT

To ensure that the Earth Observing System Data and Information System (EOSDIS) can play a role in earth science information systems that are likely to emerge in the next century, it is important that a suitable architectural direction is established from the outset of its development. The paper describes an open architectural concept under development by NASA for EOSDIS which supports site autonomy and independent, evolutionary development of components to improve the services offered to users. This concept is intended to ensure that EOSDIS' data sets and services can form part of a future international earth science system, but also offers several advantages for the future evolvability of the system itself.

1. INTRODUCTION

[1]NASA's Earth Observing System (EOS) is a long-term, multi-disciplinary research mission to study the processes leading to global change and to develop the capability to predict the future evolution of the Earth system on time scales of decades to centuries (Asrar and Dokken, 1993). The EOS Data Information System (EOSDIS) provides computing and network facilities to support the EOS research activities, including data interpretation and modeling; processing, distribution, and archiving of EOS data; and command and control of the spacecraft and instruments.

Although EOSDIS will eventually contain an enormous amount of valuable Earth science data, there are other sources of information that are essential to the study of climate change. Of critical importance are holdings of other Global Change agencies, such as NOAA, USGS, etc. and other international organizations. The Inter-Agency Working Group for Data Management for Global Change Data (IWGDMGC) are currently in the process of defining the Global Change Data and Information System (GCDIS) intended to provide linkages between data services through a common set of interoperability services. NASA is actively participating in these efforts.

In addition, there is also a growing interest by earth scientists in the possibility of developing information systems for earth science data which not only encompass the major data repositories but also enables users to take an active part in the information system, by providing data/services to the system (i.e. UserDIS). This approach seeks to encourage the scientific return from the investment in data and information systems by ensuring that the scientists are an integral part of the system.

Although NASA does not have the responsibility for developing either GCDIS or UserDIS it wants to make sure that its development of EOSDIS can support both of these evolutionary paths. This implies taking an architectural direction which opens EOSDIS so that it can be included within wider data systems and identifying architectural components which EOSDIS might contribute to these systems. The remainder of this paper discusses EOSDIS relative to GCDIS/UserDIS

This paper summarizes the results of NASA's preliminary architectural investigation, currently in progress. This paper presents high level user issues related to a generalized data and information system followed by an outline of an architectural concept for such a system. This is followed by a discussion of the major issues that would need to be resolved for the development of such a system.

[*] The work of the first two authors was funded as part of the Hughes Applied Information Systems contract to NASA for the EOSDIS Core System Project (contract NAS5-6000)

2. ARCHITECTURAL ISSUES

The nature of Global Change and earth science research in general lead to some key architectural drivers for a data and information system to support this research:

- The organizations which participate in the network are autonomous entities, and the architecture should intrude upon autonomy to a minimal extent. For example, the architecture cannot dictate how organizations will manage their data, their networks, and their users internally.

- Developing experiments, instrumentation, algorithms, and hence new kinds of data is an integral part of scientific research. Different science disciplines may have valid preferences for different or new data formats and tools. The architecture cannot make adherence to strict data interchange format standards or the use of specific tool sets a precondition of network operation.

- Scientists collaborate on research projects and exchange scientific information in many different ways. For example, scientists may want to obtain the latest set of re-calibrated satellite data or the outputs of a revised earth atmospheric model as inputs to their own analysis. The architecture should facilitate this kind of collaboration and exchange and extend to new ways of collaboration which future technologies may enable.

- A key objective of the Global Change Research Program is the establishment and maintenance of a high quality set of earth science parameters for an extended period of time which have research community consensus [from 'Global Change Data and Information System (GCDIS): A Draft Tri-Agency Implementation Plan', DOI/NASA/NOAA, March 1992]. The research community as a whole has been challenged to cooperate in the validation, upgrade, and description of this data. The architecture needs to support the research community in finding relevant data, in the analysis and critical review of this data and the publishing and dissemination of new and revised data products.

- The science expertise is distributed on a world-wide scale. The architecture should make it possible to distribute the appropriate functions and data to where the expertise resides. The cooperative endeavor envisioned by the GCRP also means that the architecture must allow researchers to take advantage of the distributed functions and data in collaborative efforts.

- Scientists would like to access scientific information in many different ways. The architecture should extend to new ways of data access which future technologies may enable. Data volume makes it impossible today to perform large scale searches on science data based on their contents, but the rapid evolution of processing and storage technology may change that in the not too distant future.

- The collaboration between geographically distributed researchers is limited by current communications facilities to file and mail exchanges. In the future, it may be possible to exchange data in real time, and view and browse them in a coordinated fashion while communicating annotations and comments. Such developments will have a profound impact on how earth science information is accessed and used for research.

- It is a characteristic of global change research that it tries to correlate information which spans long periods of time, and that experiments may go on for many years. Therefore, the architecture must be able to accommodate the long term evolution of technology and its application to science, but at the same time it must provide some measure of stability, e.g., in terms of backward compatibility.

The system architecture for EOSDIS must support both users and service providers. The concept of a service is synonymous with the concept of data, since data can only be accessed through a service (e.g. ftp, DBMS, etc.). The users of the system want to be able to find and access relevant services within the system as efficiently as possible. Service providers want to be able to support the mission objectives in terms of capturing and maintaining the important data sets, and ultimately providing the best services possible to the user community. To achieve the latter goal each provider needs the flexibility to organize their data and services in the most appropriate way for their user community and be free to re-configure existing services and add new ones to accommodate new user requirements and/or new technological capabilities.

In this scenario of autonomous service providers, there will inevitably be the potential for incompatibilities between the services available from the system and a user’s query, or the tool they are using to access the service(s). The system must recognize that incompatibilities will exist and assist the user in overcoming them as effectively as possible.

An important focus for EOS is interdisciplinary science. This will lead to user requests which cannot be resolved by a single service or even a single service provider. The system architecture must therefore support the concept of multi-site requests, which must be partitioned and managed between several services. An example of this type of request and how it might be managed is shown in Figure 2.

The main objectives of the architecture are, therefore, the definition of (a) capabilities which let a scientist locate, obtain, or use resources which are available in the network (e.g., tools and data); (b) features which would help a scientist cope with the ensuing problems, e.g., of differences in data formats, terminology, and tool input and output requirements; and (c) support functions which would make it easier for scientists to collaborate on research projects across the network.

In widening the constraints of the architectural concern from the earth observation focus of EOSDIS towards global change research, to the wider earth science focus of GCDIS and UserDIS, there are many issues which are wider in scope than if the EOSDIS requirements alone were being considered..

These are discussed in the GCDIS/USerDIS study (NASA, 1994a) and are summarized below:

- There will be considerable variety in the user objectives, missions and priorities. Users will also be data providers.
- The architecture must not assume a common information model or management system. The adoption of evolving standards should be encouraged within the GCDIS / UserDIS community, though this should be achieved through participation in the standards process by the agencies and organizations involved rather than the development of specific standards for these systems.
- There should not be any restrictions on the number of providers, their location and the data/services they provide. The system must be able to cope successfully with dynamic data and network topology.
- All responsibilities for system management or development policies and authorities will be voluntary, though within a part of the system, such as EOSDIS, can be mandated by some management authority. The architecture must therefore accommodate autonomously managed provider sites and not assume a single management approach to development, operation, user authentication or data protection. In particular the system should not depend on the availability of network wide management information.
- The data management solutions should be scalable, and cost effective to scale. The design of components should avoid limits on capacity which preclude low-end providers or restrict what high-end providers can offer.
- The architecture must help the user work effectively within an environment characterized by variability of quality in terms of responsiveness, reliability, accuracy, availability, and throughput.

Taken together, these characteristics present some significant challenges to the design and development of EOSDIS if it is to be part of a wider data system and be a major supplier of components for such a system. It is important therefore that EOSDIS establishes what it is able to achieve within its cost and schedule budgets, and leaves open to future development those aspects beyond its scope.

3. ARCHITECTURAL CONCEPT

The architectural concept for EOSDIS is shown in Figure 1 and described in further detail in NASA 1994b. It can be divided into three layers: the client layer, the service provider layer and the interoperability infrastructure. Individual sites, which may host one or more of these layers, are heterogeneous and autonomously managed. The user layer is characterized by client environments, which may be interactive (e.g., workstation graphical interface) or process environments (e.g., analysis algorithm). The interoperability layer is characterized by a set of distributed services which assess user needs against service offerings and connect the user with appropriate service providers. Finally, the provider layer is characterized by organizations who choose, or are mandated by their management authority, to provide a set of services related to data collections or to computer resources that they can offer. This includes the traditional data center concept and also specialist value-added service providers, whether commercial or government related (e.g. education specialists). Since the service provider layer must allow autonomous management and development, the details given here are limited to those which allow sites to interoperate. The architectural concept then, is in essence the interoperability infrastructure (the Intersite Architecture) and how the user and data provider services interface to this infrastructure.

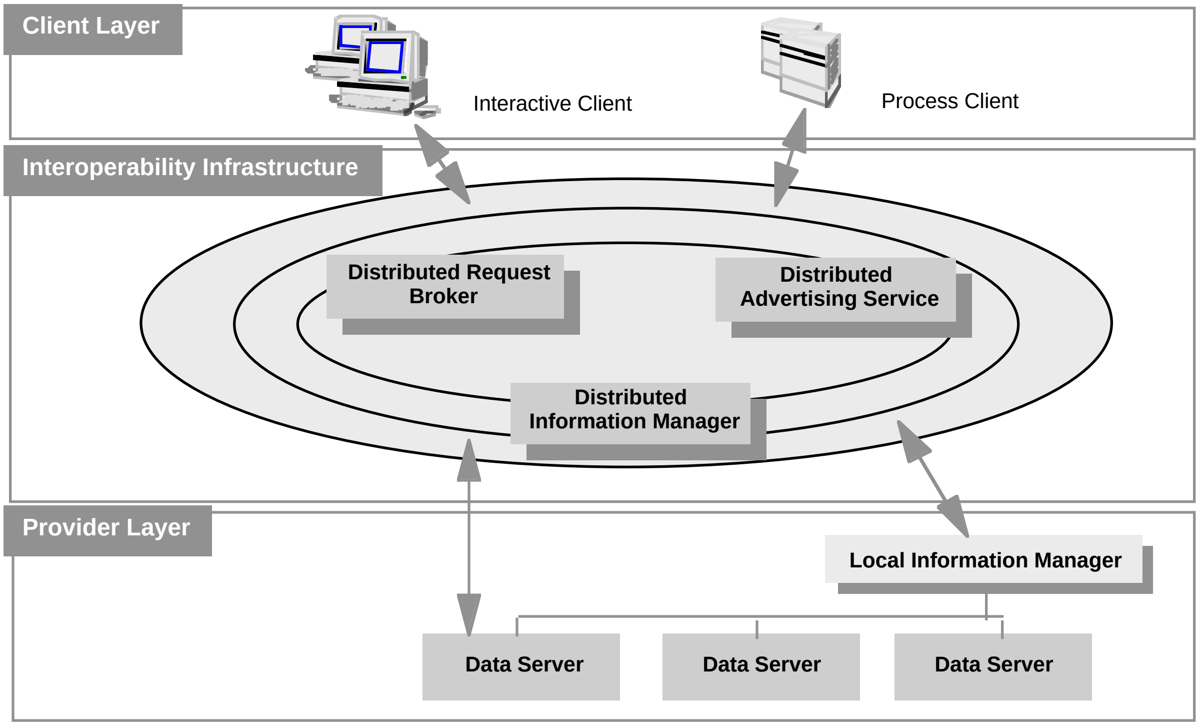
Client Layer
Interactive Client
Process Client
Interoperability Infrastructure
Distributed Request Broker
Distributed Advertising Service
Distributed Information Manager
Provider Layer
Local Information Manager
Data Server
Data Server
Data Server

Fig 1-: Conceptual Architecture Overview

3.1 Intersite Architecture

The intersite architecture is necessary to connect the services offered by providers to needs expressed by users in terms of requests to the system. Three classes of software component are envisaged for this layer:

Advertising Service: The services offered by providers on the network need to be advertised to users. The advertising service is used by other components in the intersite architecture to perform their function.

Request Broker Service: This service performs the matching between a user request and the services offered. In the many cases the user (e.g., a process) may have specified the service to be connected to, in others the broker will have to parse a 'request description' and use the advertising service to establish which services could satisfy the request. The request broking activity might involve user interaction.

Distributed Information Manager : Where multiple sites are needed to resolve a request then an Inter-site Search Service is required to manage the process. The service will break up the query if necessary and generate a plan containing sub-requests which will be processed at individual sites. The sub-requests will generally be characterized by queries and operations, where an operation is usually some manipulation of results to provide output in the form or context requested by the user. The optimization of the division of a user request into a request plan is a difficult problem and in the short-term might involve the user in the planning process.

An example of the use of the Distributed Information Manager to handle a coincident, content-based search is shown in Figure 2. In this case, the request is divided into three sub-requests: a query is made on the TOMS binned product to extract an ozone mask of the ozone hole for the required date, the mask is transferred to the second service provider which extracts cloud and sea-ice free fragments of the binned phytoplankton concentration product derived from Sea-WiFS data, which are then mosaiced on a separate compute server to form a single multi-date concentration map.

Clearly this is not a trivial problem, and represents the vision of what should be possible in the future rather than what can immediately be developed. Issues such as optimization of the query plan, incompatibility of vocabularies, etc., all need to be addressed before such a vision is achievable. However there are clear evolutionary steps on the way to the vision, each of which give the user more support for resolving science questions. The purpose of the intersite architecture is to enable this evolution without a need to leave the architecture framework.

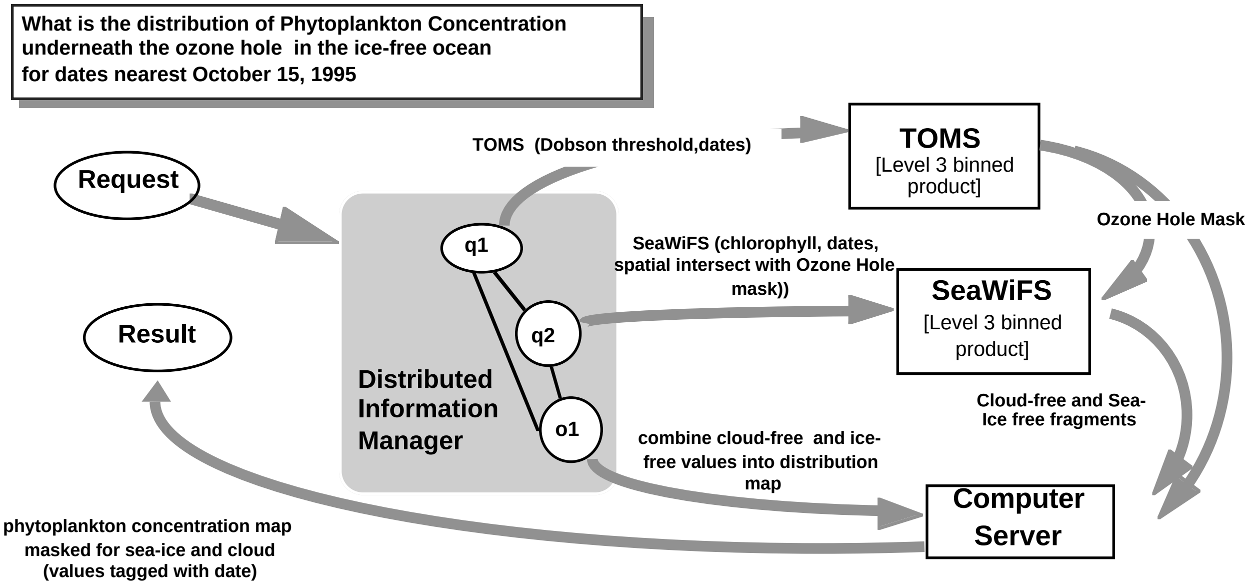
What is the distribution of Phytoplankton Concentration
underneath the ozone hole in the ice-free ocean
for dates nearest October 15, 1995
Request
Result
TOMS (Dobson threshold,dates)
TOMS
[Level 3 binned product]
Ozone Hole Mask
q1
q2
o1
Distributed Information Manager
SeaWiFS (chlorophyll, dates, spatial intersect with Ozone Hole mask))
SeaWiFS
[Level 3 binned product]
Cloud-free and Sea-Ice free fragments
combine cloud-free and ice-free values into distribution map
Computer Server
phytoplankton concentration map masked for sea-ice and cloud (values tagged with date)

Fig 2. Example - Use of Inter-Site Request Agent to Resolve Coincident Search

The main issues related to the development of these three intersite components are related to vocabulary management and mapping, service and client incompatibility management and support for multi-protocol access to a dataset. These issues are considered further in NASA, 1994a.

The interoperability layer assumes that each of its services can be distributed. There will probably be more than one example of a request broker or distributed information manager. They can differ in the scope of their capabilities or to which services and providers they give access; that is, intersite services can themselves be heterogeneous. For example, a distributed information manager might provide relatively basic inter-site request capability across a wide range of services, or more sophisticated capabilities over a smaller subset.

3.2 Provider Interfaces to the Intersite Architecture

Each provider site chooses, or is mandated by its management, to provide certain services to the system community, or some part of that community. In many cases a service would be related to data set(s) held at the providers site, but it is not mandatory; a service potentially could access data held at another site, or even provide a service for data passed to it from another site as part of a request.

The main issues to be resolved in the interfacing of a provider services to the entire system are briefly described in this section; they are:

- autonomous internal organization
- advertising services to users
- support for searching
- support for notifications to users when new data and/or services are available - subscription services
- support for incompatibility management
- the role of data servers.

Provider sites should be allowed to *autonomously organize* and manage their internal services and data to permit political and technical flexibility and therefore a definition of how the provider organizes their data and services is not part of the architectural concept. What matters is the what the site offers for external access and how it can be accessed externally. The following discussion defines how the provider architecture is represented to the system.

Sites *advertise* their services to the system; services which are not advertised do not exist from the system perspective. The advertisements are managed by the advertising service within the interoperability infrastructure and describe what the service is and how it can be accessed. Since data is essentially equivalent to a service (a user can't access data at a provider site without some sort of service), advertisements can refer to data and services.

Sites which offer *data searching* must have an external interface for accepting searches (e.g. from the Distributed Information Manager), and a service for processing these requests. This is called the Local Information Manager, and is equivalent to the Distributed Information Manager, in that it resolves inexact search requests into exact queries which can be placed on individual data servers. This might involve some interaction with the user.

Sites may let users (or programs) subscribe to data which they store or distribute. This might be used, for example, to inform a user when new data is acquired for their particular study area which exceeds a certain quality threshold. To make this available the site must offer an external interface for a *subscription service*. Users can indicate their areas of interest in a search language which are then routed like other searches to services which can monitor the subscription. These subscription services then send notifications of new information items in those areas matching the user's request whenever they are found as part of the routine processing and archiving activity (see Figure 3).

The importance of this concept is that it will free scientists from mundane information/data hunting, allowing them to specify interests periodically and then receive notifications of new data and/or services that are relevant to those interests. The acceptance of this functionality and its implementation could have a major impact on the way data and information systems are developed for the future.

The nature of the architectural concept described here will result in potential *incompatibilities* between the user tools in the client layer and the services provided. Characteristics of data, tools, services, etc. which are essential for determining incompatibility will be captured in an 'interoperability profile'. This profile is specific to the type of object under consideration (e.g. data format profile). Providers will need to adhere to common conventions for describing an interoperability profile, and they need to provide external interfaces for obtaining, exchanging and negotiating such profiles. Using these profiles it will be possible to warn users of potential incompatibilities, and offer advice on how to mitigate them.

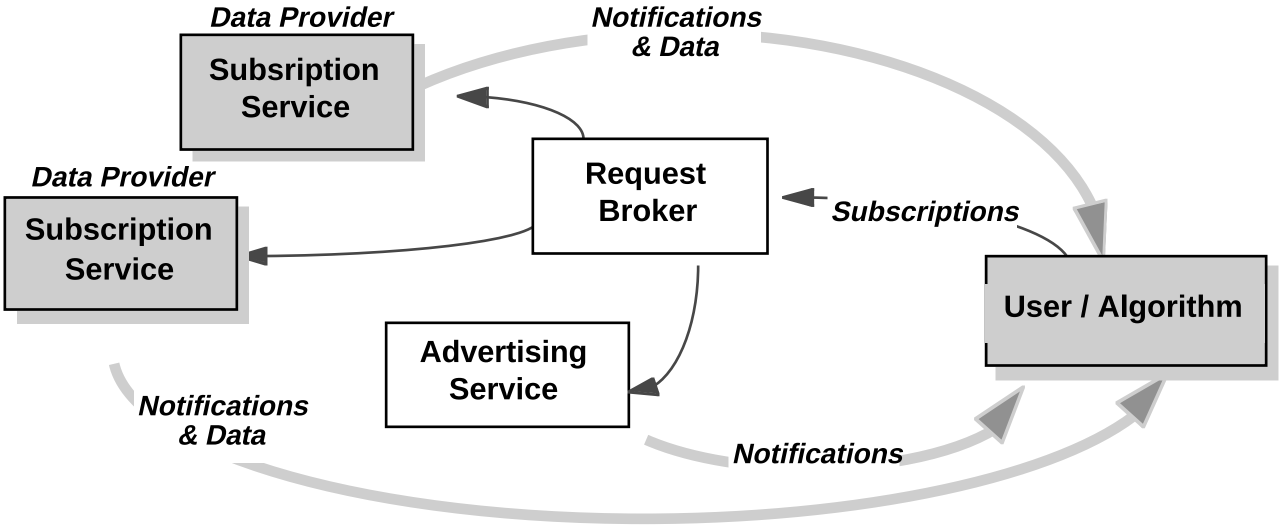

Fig 3: Subscription Service Concept

At each provider site the messages passed from the client layer to the interoperability structure need to be interpreted and acted upon. This is achieved through two routes. The most direct is a message directed at a *data server* which has knowledge of all the services related to its data set and passes the message to the correct service. At sites providing several data sets, the message might need to be interpreted against several data set servers and this would be handled by the Local Search Service. This is equivalent to the Distributed Information Manager, but only deals with the data held locally.

Data at a site is organized into one or more collections of related items. Each collection forms a data set which will contain both data and meta-data, the discrimination between these being provider and data set specific. To each data set one or more services are attached; the services may operate on all types of data in the data set or only one part of it, e.g., a relational database management system (DBMS) to an inventory table. Descriptions of these services and the parts of the data set they operate on are passed to the advertising service, establishing a 'data scope' against which users requests can be evaluated against. These services are called 'type' services in the concept since one type of service may be related to all data of that type (e.g., a text query service could be used for all text data within one data set and across all data sets at one site). Careful design of the type services should mean that they are adaptable by other providers for similar data. It is possible that a particular type of data would have more than one 'type' service associated with it, (e.g., two different text query services to support different access protocols) and that a type service would deal with more than just a single data type (e.g., an OO DBMS could deal with inventory data and the associated browse data).

3.3 User Interfaces to the Services

As for the provider layer, the architectural concept does not mandate how the user interfaces should work, only that they are compatible with one or more of the protocols that the interoperability infrastructure supports. The GCDIS / UserDIS concept seeks to encourage community development of components, and it is likely that the client layer is one of the areas where this will have the most impact.

Three categories of access to the interoperability layer are considered in the architectural concept:

general access interfaces: It is assumed that general interfaces which are applicable to a group of similar services across all data in the network (e.g. text search). This type of interface will be the main access to the services of EOSDIS and many of the large archives and will be similar in concept to the present day EOSDIS Version 0 and ESA's User Interface Terminal (Simpson *et al.*, 1993) interfaces. The general interfaces will be customized in their operation by specific information from the service provider (e.g. vocabulary), etc.

specialized access interfaces: The above approach would support dynamic modification of an existing interface, but would not support special interfaces for specific services, e.g. an interface which is particularly oriented towards the coincident location and analysis of sea surface temperature 'images' and sub-surface profile measurements of temperature and salinity. In this case the service provider might be able to provide a software module which would be dynamically linked into the user's interface and provide a completely specialized interface, configured to a specific service/provider.

object access interfaces: Finally, objects resulting from previous queries should be capable of initiating further service requests. For example a search of an image inventory might result in a results object which contains the inventory records matching the query and a reference in the object which would enable a user to automatically initiate a browse service and review the image being referenced by one of the inventory records.

4. DEVELOPMENT ISSUES/APPROACH

The EOSDIS architectural concept described here offers several important advantages for some future development of GCDIS and/or UserDIS. First, data providers have complete freedom of choice as to how they wish to organize data into types. They may want to use EOSDIS provided type services or data type servers, or they may decide to create their own and link them into the local search and retrieval service software. They may even decide to replace all data types with a fully integrated database management system (and appropriate schema); several research projects and commercial ventures in the area of object databases are in progress and may mature during the lifetime of the global change program (Stonebraker and Rowe, 1986; Haas *et al.*, 1990; Lamb *et al.*, 1991).

Secondly, the concept supports the inclusion of legacy systems into the network. A site is only required to advertise only those services it wishes to support in the GCDIS/UserDIS context; there is no minimum set of services which a site must provide. For example, a site may only be able to offer text search and file transfer, but would still be able to contribute to the network.

Thirdly, the architecture provides an open ended approach to earth science data search and retrieval. Searches can be manipulated at the level of the intersite search service, local information manager service, or at the level of the data type server. Search capabilities can be negotiated among services and do not enter into the protocols themselves. This permits the future application of results of ongoing research, e.g., in areas of query and schema translation, query optimization, and search languages might be adaptable to earth science data types (Ordille and Miller, 1993; Morrissey, 1990). For example, future search engines may be able to assist users through intelligent searching using knowledge bases created by the earth science community, and data providers could then advertise "knowledge" about their data holdings rather than indexes (Smith *et al.*, 1989).

Fourth, the architecture makes no principal distinction between various levels of earth science metadata (e.g., directory versus inventory), or between metadata and data. This is in realization that, in a GCDIS and UserDIS environment, different organizations may very well have differing concepts for these objects. The widely differing data holdings at these organizations also will likely lead to differing interpretations of what is index, what is data, and what can be searched within a reasonable time. Despite this variability in data it is important that searches on multiple data sets provide results which can be compared effectively by the user, and thus the query process will include mechanisms to ensure that the user receives the required form of result. The architectural approach shown facilitates the introduction of more powerful search strategies in the future (Hellerstein and Stonebraker, 1993; Haas, 1989).

Finally, the concept described above will encourage evolutionary and independent development of system components. By adopting a fully distributed architecture for all components and not mandating the details of the client interface and service implementations, the entire user and development community can participate in the development of components in each of the three layers. For example, computer science research may lead to the development of an improved intersite search agent. Users can then choose whether the new agent provides a 'better' service. If it does then, over time, it will make other agents obsolete. Moreover by establishing a conceptual framework which can accommodate the variability of the earth science discipline which can guide rather than constrain development of components, hopefully minimizing the 'not invented here' syndrome, it will encourage the development of components and support utilities (e.g. APIs) by the entire community.

Although the architectural concept seeks to strike a proper balance between the users' demand for decentralized capabilities and autonomy on one side, and complete anarchy on the other, a network of the type proposed for GCDIS /

UserDIS poses significant issues in several system quality areas. For example, the accuracy of search results suffers as incompatibilities among the vocabularies and terms employed by different data providers increases. In an unmanaged network, there can be no expectations regarding service reliability, availability and response time. For example, some sites may respond to a search within seconds or minutes, others may not respond for days because the data provider experiences hardware problems.

The solutions to these types of problems are outside the scope of an architecture. They depend on the cooperation of service providers which, in a network like UserDIS, is voluntary. However, the architecture can include measures to facilitate the solutions. For example, EOSDIS will not make a reliable network, in which all sites are always available, a precondition for successful operation. The services will provide feedback which lets users judge the quality of a response (if they so desire). The architecture will provide mechanisms for characterizing situations where standards or conventions exist and are being followed.

As described above there are several areas where the computer science community could contribute solutions to the GCDIS and UserDIS challenges. In each area EOSDIS will need to pick specific technical approaches which are compatible with its implementation time frame, while encouraging the computer science community to seek improved solutions which can replace the baseline approach in the future.

5. SUMMARY

The GCDIS / UserDIS concept describes a radical departure from the traditional model of data system. By taking this concept into consideration in its development of EOSDIS, NASA will provide some components of a system in which an open interoperability standard can be used to acquire or provide data and services, enabling an information system to be developed that will operate more as a marketplace with positive competition than as a monolithic, monopoly that focuses on production and storage of data.

Such an information system should encourage evolutionary and independent development within a single framework on an inter-agency and international scale. Indeed its success depends on this complementary development. It should also provide more flexibility for accommodation of new user needs and taking advantage of emerging technological developments. Finally, it provides more flexibility to respond to the inevitable change in distribution, prioritization and funding policies over such a long-term undertaking as an earth science information system.

ACKNOWLEDGMENTS

This work was undertaken as part of Hughes Applied Information Systems contract to NASA for the EOSDIS Core System Project (contract NAS5-6000). Important recommendations which led to the approach described in the paper were received from the National Research Council’s review of the EOSDIS plans.

REFERENCES

Asrar, G. and D. J., Dokken, 1993. EOS Reference Handbook, NASA, March 1993.

Haas, L.M., Freytag, J.C., Lohman, G.M., and P. Pirahesh, 1989. "Extensible Query Processing in Starburst," Proceedings of the ACM SIGMOD'89, June 1989.

Haas, L.M. *et al.*, 1990. "Starburst Mid Flight: As the Dust Clears," IEEE Transactions on Knowledge and Data Engineering, March 1990.

Hellerstein, J.M. and M. Stonebraker, 1993. "Predicate Migration: Optimizing Queries with Expensive Predicates", Proceedings of the ACM SIGMOD '93, June 1993.

Lamb, C., Landis G., Orenstein J., and D. Weinreb, 1991. "The ObjectStore database system," Communications of the ACM, Vol 34, Number 10, October 1991.

Morrissey, J.M., 1990. "Imprecise Information and Uncertainty in Information Systems," ACM Transactions on Information Systems, Vol 8, No 2, April 1990.

NASA, 1994a. GCDIS/UserDIS Study, EOSDIS Core System Project, in preparation.

NASA, 1994b. ECS Science Information Architecture, Working Paper FB9401V2, March 1994.

Ordille, J., and B. Miller, 1993. "Database Challenges in Global Information Systems," Proceeedings of the ACM SIGMOD'93, June 1993.

Smith, P.J., Shute, S.J., and D. Galdes, 1989. "Knowledge-Based Search Tactics for an Intelligent Intermediary System," ACM Transactions on Information Systems, Vol 7, No 3, July 1989.

Stonebraker, M. and L. Rowe, 1986; Stonebraker, L. Rowe, "The Design of Postgres", Proceedings of the ACM SIGMOD'86, June 1986.